AMENDMENTS IN THE CLAIMS

1.   (presently amended) A method for keeping files current for use in a <u>client</u> computer system coupled to a network, <u>the method</u> comprising the steps of:

evaluating <u>at said client</u> a downloaded file from a source within said network to determine if a source identifier is present in said downloaded file;

checking said source periodically utilizing said source identifier to determine if a newer version of said downloaded file exists; and

replacing <u>at said client,</u> in response to the presence of said newer version of said downloaded file, said downloaded file with said newer version.

2.   (presently amended) The method as recited in claim 1 wherein said step of evaluating further includes the step of attaching, in response to ~~said~~ <u>no</u> source identifier ~~not~~ <u>being</u> present, a source ~~descriptor~~ <u>identifier</u> to said downloaded file.

3.   (original) The method as recited in Claim 1 wherein said step of replacing said downloaded file includes the steps of:

providing an indication to a user that said newer version of said file exists;

prompting said user to replace said downloaded file with said newer version; and

replacing, in response to said user requesting said newer version, said downloaded file with said newer version.

4.   (original) The method as recited in Claim 1 wherein said source identifier is located in the extended attribute of said downloaded file.

5.   (presently amended) The method as recited in Claim 1 wherein said downloaded file ~~is selected from the group consisting of PDF and ZIP files~~ <u>comprises one or more stored parameters from among: (1) a signature string utilized to find said source identifier within said file; (2) a locator string identifying a location from which the file is sourced; (3) a date/time and version number of said file; and (4) a checksum string covering prior entries of said file.</u>

6.   (original) The method as recited in Claim 1 wherein said source identifier is an uniform resource locator (URL).

7.   (presently amended) The method as recited in Claim 1 wherein said step of checking said source periodically includes:
       defining a <u>default, automatic</u> time interval <u>at which said checking step is initiated; and</u>
       <u>enabling a user to adjust said time interval, if desired</u>.

8.   (presently amended) The method as recited in Claim ~~7~~ <u>1</u>, wherein said ~~time interval is user-defined~~ <u>replacing step further comprises</u>:
       <u>renaming a previously stored copy of said downloaded file on said client system from a current working name to an archived name; and</u>
       <u>storing said newer version of said downloaded file with the current working name of the downloaded file</u>.

9.   (presently amended) The method as recited in Claim 1 wherein said step of checking said source <u>comprises checking said source</u> ~~is~~ whenever said downloaded file is opened<u>, wherein, when said checking step includes a defined periodic time interval at which said checking is automatically initiated, said method further comprises overriding said time interval by initiating said checking step whenever said downloaded file is opened</u>.

10.   (presently amended) The method as recited in Claim 1<u>, further comprising storing an identifier and a source descriptor of said downloaded file and each newer version of said downloaded file in a specially coded file registry, which is checked by a controller for correct file location during said checking step</u> ~~wherein said step of checking said source is initiated by a user~~.

11.   (original) The method as recited in Claim 1 wherein said network is a packet network.

12.   (original) A computer system for use in a network environment, comprising:
       a processor;

an update manager coupled to said processor, including:

means for evaluating a downloaded file from a source within said network to determine if a source identifier is present in said downloaded file;

means for checking said source periodically utilizing said source identifier to determine if a newer version of said downloaded file exists; and

means for replacing, in response to the presence of a newer version of said downloaded file, said downloaded file with said newer version.

13.   (presently amended) The computer system as recited in Claim 12 wherein said means for evaluating further includes means for attaching, in response to ~~said~~ no source identifier ~~not~~ being present, a source ~~descriptor~~ identifier to said downloaded file.

14.   (original) The computer system as recited in Claim 12 wherein said means for replacing said downloaded file includes:

means for providing an indication to a user that said newer version of said file exists;

means for prompting said user to replace said downloaded file with said newer version; and

means for replacing, in response to said user requesting said newer version, said downloaded file with said newer version.

15.   (original) The computer system as recited in Claim 12 wherein said source identifier is located in the extended attribute of said downloaded file.

16.   (presently amended)   The computer system as recited in Claim 12 wherein said downloaded file ~~is selected from the group consisting of PDF and ZIP files~~ comprises one or more stored parameters from among: (1) a signature string utilized to find said source identifier within said file; (2) a locator string identifying a location from which the file is sourced; (3) a date/time and version number of said file; and (4) a checksum string covering prior entries of said file.

17.   (original) The computer system as recited in Claim 12 wherein said source identifier is an uniform resource locator (URL).

18.   (presently amended) The computer system as recited in Claim 12 wherein said means for checking said source periodically includes:
        defining a default, automatic time interval at which said checking step is initiated; and
        enabling a user to adjust said time interval, if desired.

19.   (presently amended) The computer system as recited in Claim 18 wherein said ~~time interval is user defined~~ replacing step further comprises:
        renaming a previously stored copy of said downloaded file on said client system from a current working name to an archived name; and
        storing said newer version of said downloaded file with the current working name of the downloaded file.

20.   (presently amended) The computer system as recited in Claim 12 wherein said means for checking said source comprises checking said source ~~is~~ whenever said downloaded file is opened, wherein, when said checking includes a defined periodic time interval at which said checking is automatically initiated, said system further comprises means for overriding said time interval by initiating said checking whenever said downloaded file is opened.

21.   (presently amended) The computer system as recited in Claim 12 ~~wherein said means for checking said source is initiated by a user,~~ further comprising means for storing an identifier and a source descriptor of said downloaded file and each newer version of said downloaded file in a specially coded file registry, which is checked by a controller for correct file location during said checking step.

22.   (presently amended) The computer system as recited in Claim 12 wherein said network is a packet network and said computer system is a client system coupled to said network.

23.   (presently amended) A computer program product comprising:

a computer-readable medium having stored thereon computer executable instructions for implementing a method for keeping files current for use in a <u>client</u> computer system coupled to a network, said computer executable instructions when executed, perform the steps of:

evaluating <u>at said client</u> a downloaded file from a source within said network to determine if a source identifier is present in said downloaded file;

checking said source periodically utilizing said source identifier to determine if a newer version of said downloaded file exists; and

replacing <u>at said client</u>, in response to the presence of said newer version of said downloaded file, said downloaded file with said newer version.

24.   (presently amended) The computer program product as recited in Claim 23 wherein said step of evaluating further includes the step of attaching, in response to ~~said~~ <u>no</u> source identifier ~~not~~ <u>being</u> present, a source ~~descriptor~~ <u>identifier</u> to said downloaded file.

25.   (original) The computer program product as recited in Claim 23 wherein said step of replacing said downloaded file includes the steps of:

providing an indication to a user that said newer version of said file exists;

prompting said user to replace said downloaded file with said newer version; and

replacing, in response to said user requesting said newer version, said downloaded file with said newer version.

26.   (original) The computer program product as recited in Claim 23 wherein said source identifier is located in the extended attribute of said downloaded file.

27.   (presently amended) The computer program product as recited in Claim 23 wherein said downloaded file ~~is selected from the group consisting of PDF and ZIP files~~ <u>comprises one or more stored parameters from among: (1) a signature string utilized to find said source identifier within said file; (2) a locator string identifying a location from which the file is sourced; (3) a date/time and version number of said file; and (4) a checksum string covering prior entries of said file.</u>

28.   (original) The computer program product as recited in Claim 23 wherein said source identifier is an uniform resource locator (URL).

29.   (presently amended) The computer program product as recited in Claim 23 wherein said step of checking said source periodically includes:

defining a default, automatic time interval at which said checking step is initiated; and

enabling a user to adjust said time interval, if desired.

30.   (presently amended) The computer program product as recited in Claim 29 wherein said ~~time interval is user defined~~ replacing step further comprises:

renaming a previously stored copy of said downloaded file on said client system from a current working name to an archived name; and

storing said newer version of said downloaded file with the current working name of the downloaded file.

31.   (presently amended) The computer program product as recited in claim 23 wherein said step of checking said URL comprises checking said source ~~is~~ whenever said downloaded file is opened, wherein, when said checking step includes a defined periodic time interval at which said checking is automatically initiated, said method further comprises overriding said time interval by initiating said checking step whenever said downloaded file is opened.

32.   (presently amended) The computer program product as recited in Claim 23 ~~wherein said step of checking is initiated by a user~~, further comprising storing an identifier and a source descriptor of said downloaded file and each newer version of said downloaded file in a specially coded file registry, which is checked by a controller for correct file location during said checking step.

33.   (original) The computer program product as recited in Claim 23 wherein said network is a packet network.